

EXTENDED BASIC LANGUAGE has been mentioned in these pages for quite a while now, and we are finally able to state that it is available! Jay Fenton, developer of the Bally Basic and the new AstroVision Basic, called me about two years ago, asking if there was any interest in a new language he was then working on. I told him it would have to be able to address more memory space than the 1800 bytes of Bally Basic, and he went back to the drawing board. Subsequently, as he developed the language, and Perkins Engineering developed the Blue Ram memory addition, and Alternative Engineering developed the Viper System memory addition, we were instrumental in having these gentlemen talk to each other to ensure that compatibility would not be precluded. The first tapes of the new language have been received by Alternative Engineering, and they will be supplied free with the smallest expansion they provide, the System 1. See their ad on p. 89. The language will also be provided in a ROM form for direct plug-in into the Arcade, by Perkins Engineering. The engineering work to accommodate this form of presentation is now underway. I have received a sample of the SYSTEM 1, and the Extended Basic language. The language is a great step towards ZGrass in that it has the graphics features of POINT, CIRCLE, SNAP, SHOW, and WINDOW, two sizes of lettering, plus the capability to have four colors anywhere on the screen.

BASIC MANIPULATION One of the things that we plan to do is create small modifications to the Extended Basic language, to provide optional or personalized features. These options could take the form of short programs to be loaded after the Basic was loaded, and it would modify the language itself. We currently modify programs to do things we want, this opens up the language to changes. Conceivably, one could develop a "floating point arithmetic" package modification, or scientific math applications, or new graphic characters for a game or other application.

DELAYS are a familiar story to all-time subscribers. The new AstroVision Basic cartridge was pulled from production in order to make some corrections. George Moses and Tom Wood each found a "glitch" that could cause the program to reset under special circumstances. Even though those circumstances would rarely occur, the AstroVision people decided to make the corrections needed, take the delay in production, but end up with a better product. As it is, about a month was lost. Those of you who have ordered the "AstroVision Basic Listing" will find an error note on page 23 of the listing, covering this.

WHAT TO PRINT or sometimes what not to print, is perplexing. We cannot restate all the basics in each issue, even in each volume. Yet new and unsophisticated readers join us each month, and wonder what we are talking about. My original intent in setting up the Volume structure was to gradually carry us all from the rudiments on towards the sublime. In other words, to get the full story, one has to have all the material from Day One since we can't reinvent the wheel too frequently. The basic tutorial material is available in Volumes 1 and 2, as well as numerous good programs, and each Volume is available at \$10.

GRAPHICS and how to produce them is always of interest, and we have an article by Rich Tietjens in this issue, explaining character generation. Coincidentally, the June issue of Creative Computing has a large article discussing how the TV makes graphics, with the use of the Atari computer. The generalities are the same for all computers, so that this would be an excellent adjunct to Rich's article.

ARCADIAN

PRE-TUTORIAL

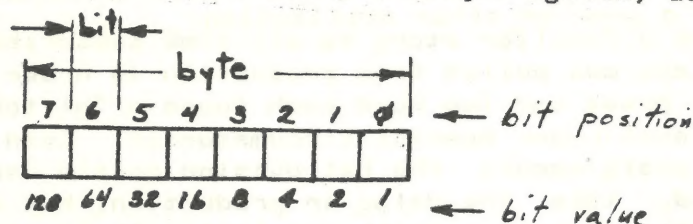
The "pixel" is the smallest object that can be shown on the screen. Its actual size is determined by two things, the physical size of the TV screen itself, and the "resolution" that the computer is capable of. The Arcade will divide every screen into 16320 pixels, disposed at 102 high by 160 wide. Obviously, the pixel is therefore bigger on a bigger screen. These 16320 pixels form the "low resolution mode", and by adding more memory (plus a few other items), the resolution can be increased to twice as many (medium resolution) or twice again to 49280 pixels (high resolution). The Perkins mod described on page 63 can provide any of these resolutions. Well, with four times as many pixels on the same screen, each one has to be one-fourth the size of the original, so detail of objects, characters, etc., is greatly enhanced.

Each pixel is turned "on" or "off" by the TV's electron gun each time it makes a sweep of the screen, so you have to have enough memory on board to remember the status of each pixel. The screen is therefore "refreshed" at every sweep.

The computer stores such common things as characters, letters, etc., in little subroutines and "calls" each one as requested by the program, without having to generate the character every time. The program in Rich's tutorial below indicates a simplified way for you to generate your own character (for a game, etc.,) and control its location by setting up a subroutine that your big program would call as needed. To do this, he sets up a 16 pixel by 10 pixel workspace, and determines which of the pixels in that space has to be "on" to generate the character. This is done outside the program, but the answers are entered into the program for permanence.

BITS, BYTES, and PIXELS

There are eight "bits" in a "byte". The computer uses bytes in most of its computations, and each one has a number assigned to it, from 0 to 255. This value is calculated by a unique system, as shown:



Each bit can be either a "1" or a "0". For each "1" that is shown, one adds together all the corresponding "bit values". This sum is the "byte value". Every number from 0 to 255 can be developed this way, and there is only one combination to do it. As an example, for a byte value of 84, bits number 6, 4, and 2 must be set equal to "1", for their bit values are 64, 16, and 4 respectively, their sum is 84. Mathematicians will see that the bit values represent the powers of two raised to the bit position.

Now to pixels. Each bit is equivalent to a pixel insofar as the screen picture is concerned. At every "1", the screen will be "on", so we can easily generate a picture by selecting the bits that we want to be "on", determining the byte value, and telling the computer that information - as Rich does in the following:

TUTORIAL Creating Special Graphics, by Rich Tietjens

Most of us who program in Bally BASIC have often wished for a faster means of creating complex characters on the screen. The most obvious method uses a machine-language subroutine, but many hackers have no desire to "get involved" in the complexities of this mysterious and arcane art. However, if you are willing to do some doodling on paper and a little basic arithmetic, you can create your own special characters without headaches.

First, you must write (or modify) your program to employ the subroutine. This requires at least two variables to position the character; I chose H (for Horizontal) and V (for Vertical).

Now, wherever you want to draw your character, you must set the H and V values; the position 0,0 is in the upper left corner of your screen. Position H,V will define the upper left corner of a character block 16 pixels wide by 10 pixels high. H may range from 0 to 159; V from 0 to 99 (which is off the bottom of the screen). Then POKE 20203, Vx256+H ((that is, program the statement to read - -(20203)=Vx256+H)) Now, CALL 20200 Voila! There is your character! But wait, we can't RUN your program without the subroutine in place; so make your changes; then go ahead with the rest of this story.

The following routine will draw, move, and limit to the screen area a special character.



When your main BASIC program is modified to your satisfaction, save it to tape; verify a good recording and set it aside.

Now let's define your characters; each is drawn on a 16x10 matrix, thusly:

	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0																
1					X	X	X									
2				X	X	X	X	X								
3			X		X		X		X							
4				X	X	X	X	X								
5					X	X	X									
6				X	X	X	X	X								
7			X		X	X	X		X							
8		X								X						
9	X	X	X							X	X	X				
	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1

Note that this positions the figure to the left of the available 16 pixel block. As a drill, see if you can determine the entries that will place the figure at the right side within the block.

RICH TIETJENS
501 S. ARCHER #5
SAN ANGELO, TEXAS 76903

```

10 . POSITION AND DRAW
20 -(20203) = 256 b V + H; CALL 20200
30 . GET NEW POSITION
40 X = H + JX(1); Y = V - JY(1)
50 . WRAPAROUND
60 IF X > 159 X = 0
70 IF X < 0 X = 159
80 IF Y > 99 Y = 0
90 IF Y < 0 Y = 99
100 . ERASE
110 CALL 20200
120 . CHANGE POSITION AND GO DO IT AGAIN
130 V = Y; H = X; GOTO 20
    
```


In this example, representing an Apollo Lunar Module, the X's show the pixels which will be turned on when the subroutine is called. Since no pixels are turned on in row 0, both bytes equal zero. In row 1, bits 3, 2, and 1 of the left byte are turned on, equalling 8 plus 4 plus 2, or 14; the right byte is still zero.

When you have all the values figured out, load PROGRAM A; RUN it and enter the values computed in the byte value table. If you have two characters, enter any number except zero when asked "two figures?"; enter 0 if you only have one graphic character.

Now, the program will ask "READY?". Press any key (except HALT) to have your character displayed (both will display if you have two.)

If it (they) is/are ok, put the main program tape back in the recorder, with the tape positioned just past the trailing end of the BASIC listing. Start the recorder and press "GO". PROGRAM A will assemble your subroutine and put it on tape. When it is done, you can RESET the memory, load the tape just created, and there is your pet graphic character! If you wish to switch to the second graphic character, just POKE 20213, 129 (that is, $\%(20213)=129$). POKE 20213, 128 to switch back.

If instructions are included on the tape, using the REM (.) statement, the machine language subroutine should be recorded after the instructions; otherwise the remarks will override the subroutine. And that's all there is to creating and using your own graphic characters!

(Note for Blue Ram owners: you can do the same thing, but have a lot more room to play around in. Some addresses within PROGRAM A must be changed, along with the CALL address.)

Special thanks to Tom Wood for the effort expended on the On-Board ROM subroutines.

```

PROGRAM A
_001 . GRAPHIC CHARACTER MAKER
1002 . BY RICH TIETJENS
1010 GOTO 1030
1020  $\%(Y) = V$ ;  $Y = Y + W$ ; RETURN
1030 CLEAR ; NT = 1; X = 0; Y = 20200; R = Y; W = 2; Z = 1020
1040 FOR U = X TO X + 19 STEP 2
1050 CY = 32; PRINT #2, "ROW", U c 2, ":
1060 INPUT "LEFT BYTE?" @(U)
1070 INPUT "RIGHT BYTE?" @(U + 1)
1080 NEXT U
1090 IF U < 21 INPUT "2 FIGURES?" T; IF T K = 20; GOTO 1040
1100 V = -43; GOSUB Z; V = 6965; GOSUB Z
1110 V = 10240; GOSUB Z; V = 20210; GOSUB Z
1120 V = -13871; GOSUB Z; V = -1936; GOSUB Z
1130 V = -32690; GOSUB Z; V = 12288; GOSUB Z
1140 V = 2432; GOSUB Z; V = 527; GOSUB Z
1150 V = -247; GOSUB Z; V = 78; GOSUB Z
1160 W = 1; Y = Y - W; FOR S = 0 TO U
1170 V = @(S); GOSUB Z; NEXT S
1180 PRINT "READY"; K = KP; CLEAR
1190  $\%(20203) = 0$ ; CALL R
1200 IF X > 21  $\%(20203) = 75$ ;  $\%(20213) = 129$ ; CALL R
1210 CY = 0; PRINT "OK?"; IF KP # 13 GOTO 1210
1220 :PRINT ; NT = 2; PRINT ; PRINT "CLEAR"; CY = -32
1230 FOR X = R TO Y STEP 2
1240 PRINT #5, " $\%($ ", X, " $) =$ ", #6,  $\%(X)$ 
1250 NEXT X; PRINT ":RETURN; RUN
1260 :RETURN; .END

```

don't
forget,
b means x
c means ÷

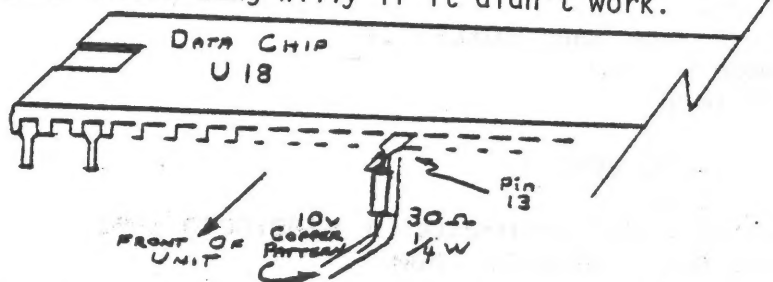
One of the modifications that have been incorporated into the AstroVision Arcade motherboard is the reduction in the power supply voltage. Dan Koppen is doing roughly the same thing to his Bally board...

Krazy Koppen's Heat Sink

Does your machine poop out after an hour of intense game playing? Do football players get mysteriously tackled by strange dots on the screen? Or does your game just stop as you are shooting your last space invader out of the sky? Well Krazy Koppen just might have the cure for you!!

The cause of all your woes is the data chip (U18). This little babe gets so hot you can fry eggs on it and what's worse, it sits right under the keyboard which makes ventilation nearly impossible. After trying everything short of installing an attic fan, a friend of mine (who shall remain nameless (after all why should I share the credit)) claimed this chip could run on much less than the 10 volts supplied to it.

With trembling fingers I wired a 30 ohm $\frac{1}{4}$ watt resistor on pin 13, and son-of-a-gun, whoosh-bang-willy if it didn't work.



The above drawing shows exactly how I did it, with one word of caution. Don't over heat pin 13 when soldering on the resistor. You may damage the data chip and have to replace it.

I have had great luck with this fix and now can leave my machine on indefinitely. Hope you have the same results!

Dan Koppen
(804)-484-1907
Suffolk, Va.


```

2 .
3 .
4 :RETURN
5 .NIM
6 .R HILFERDING
10 CLEAR ;GOSUB 9000
20 NT=0;M=15;K=0;J=0;F=0;CLEAR
30 &(9)=300;FC=0;BC=117;CY=0;CX=-49;INPUT "1 OR 2 PLAYERS?"X
40 IF (X<1)+(X>2)GOTO 30
50 IF X=2GOTO 150
60 CLEAR ;CY=0;INPUT "DEGREE OF DIFFICULTY:1-3?"Y
70 IF Y<1PRINT "THAT'S CHEATING!";GOSUB 7000;GOTO 60
80 IF Y>3PRINT "ITS HARD ENOUGH AT 3!";GOSUB 7000;GOTO 60
90 IF Y=1R=1
100 IF Y=2R=RND (2)
110 IF Y=3R=2
120 CLEAR ;CY=0
150 CLEAR ;&(9)=300;BC=7;FC=96;PRINT " WHO WANTS TO GO FIRST?"
160 PRINT "1-PLAYER 1
170 IF X=2INPUT "2-PLAYER 2"Z;GOTO 190
180 INPUT "3-COMPUTER"Z
190 IF (Z<1)+(Z>3)PRINT "IS THIS TO TOUGH?";GOSUB 7000;GOTO 150
200 IF Z=2IF X=1GOTO 150
210 IF Z=3IF X=2GOTO 150
215 H=Z;IF Z=3H=1
220 CLEAR ;FC=0;BC=134;GOSUB 5000
230 IF Z=3GOTO 4000
1000 CY=0;PRINT " PLAYER #",#1,H
1010 P=KN(H)*127+2
1020 CY=-10;PRINT "HOW MANY BOXES?",P
1030 IF TR(H)=0GOTO 1010
1033 IF P>MGOTO 1020
1040 M=M-P
1050 IF X=1IF M=0GOTO 8000
1060 IF M<=0GOTO 8500
1070 GOSUB 6000;IF X=2IF H=1H=2;GOSUB 5500;GOTO 1000
1080 IF H=2H=1;GOSUB 5500;GOTO 1000
1090 IF X=1GOSUB 5500;GOTO 4600
4000 CY=0;CX=-50;PRINT "MY TURN
4010 P=1;M=M-P;R=R-1
4020 GOSUB 6000;GOSUB 5500
4030 GOTO 1000
4600 IF P<=RGOTO 4700
4610 R=R+4
4700 IF M=1GOTO 8300
4701 R=R-P
4710 IF RGOTO 4800
4720 R=4;GOTO 4000
4800 P=R;R=4;M=M-P;CY=0;CX=-50;PRINT "MY TURN
4810 GOSUB 6000;GOSUB 5500;GOTO 1000
5000 FOR T=-75TO 65STEP 10;BOX T,35,5,5,3;NEXT T;T=70;RETURN
5500 BOX 0,0,160,34,2;RETURN
6000 FOR A=1TO P
6005 BOX T-5,35,5,5,3
6010 FOR B=35TO -44STEP -8;BOX T,B,5,5,3;BOX T,B,5,5,3;NEXT B
6020 T=T-10;IF T=-75GOTO 8300
6030 NEXT A;CY=22;CX=35;PRINT #1,M," LEFT";RETURN
7000 FOR Q=1TO 500;NEXT Q;RETURN

```

ARCADIAN

```

8000 CLEAR ;NT=3;CY=0;PRINT " I BEAT YOU!";GOSUB 7000
8010 F=F+1;GOTO 8600
8300 CLEAR ;NT=3;CY=0;PRINT " YOU BEAT ME!";GOSUB 7000
8310 K=K+1;GOTO 8600
8500 CLEAR ;NT=3;CY=0;PRINT " PLAYER #",#1,H," LOSES!";GOSUB 7000
8510 IF H=1J=J+1
8520 IF H=2K=K+1
8600 CLEAR ;FC=134;BC=96;PRINT ;PRINT "      SCORE:"
8610 M=15;PRINT ;PRINT "PLAYER #1",
8620 IF X=1PRINT "      COMPUTER
8630 IF X=2PRINT "      PLAYER # 2
8635 PRINT
8640 PRINT K,;IF X=1CX=10;PRINT F;GOTO 8660
8650 IF X=2CX=10;PRINT J
8660 PRINT ;PRINT "SQUEEZE TRIGGER TO REPLAY
8665 CX=0;PRINT "OR
8670 PRINT "PRESS '0' FOR A NEW GAME
8680 IF &(22)=16GOTO 20
8690 IF TR(1)=0GOTO 8680
8695 IF TR(1) IF X=1NT=0;GOTO 60
8700 CLEAR ;NT=0;GOTO 150
9000 FC=134;BC=0;&(10)=140;@(1)=32;@(2)=78;@(3)=73;@(4)=77;@(5)=33
9010 CY=-30;FOR A=1TO 5;TV=@(A);NEXT A
9020 NT=0;FOR C=-80TO 40;FOR D=-33TO -27
9030 IF PX(C,D)BOX (C+81)64-79,(D+30)64,4,4,1
9040 NEXT D;NEXT C
9050 BOX -59,-30,42,9,2;&(10)=174;RETURN
>

```

Robert Hilferding
P. O. Box 1647
Brandon, FL 33511

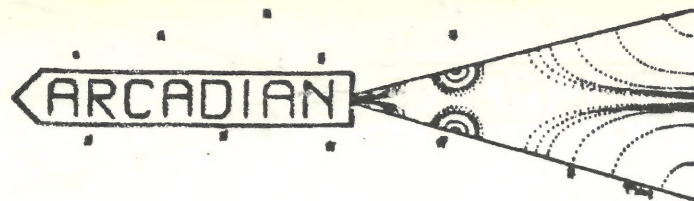
NIM is a one or two player game where the object is to force your opponent into taking the last piece. Starting with fifteen boxes, the players alternate in removing one, two, or three boxes. (KN determines the number, TR removes the boxes.) In the one player mode, the computer determines how many boxes it will remove.

The program permits options in who starts, and the degree of difficulty option in the one-player mode. Beginning options are entered through the keypad while play options are executed through the hand controllers.

Color Organ/Light Show:
Interface Bally to your Stereo
TV displays Color & Light in
step with the music.
Complete interface unit with Cassette Demo. &
Tutorial...\$29.95 (Add your own graphics/Art.)

**20 Games are now available All Use Full
Memory & Display Lots of Graphics, Action,
Sound & Color..Any Two on Cassette...\$10.00**

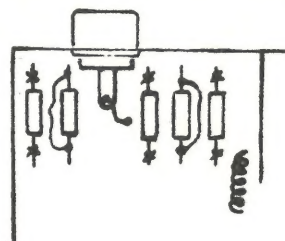
Write For Free Picture Catalog
Visa & Master Charge Accepted
L & M Software
8599 Framewood Dr.
Newburgh IN. 47630
We Pay Postage!



BALCHECK is the name of a program that is used to check out the operation of the motherboard at various times during the manufacture of the Bally Arcade. In the original advertising literature, a Videocade with this self-checking ability was to have been made available, but it was never done as far as I have been able to determine. But by George, one looks to be upon the very near horizon. The program (available at \$6.50) has been successfully entered into a PROM, and the plan is to make up a BALCHECK device for sale at a reasonable cost. The unit will connect to the 50-pin connector at the rear, and it is also very possible to package it into a cartridge-sized box so that it will slip into the game slot. This device will then analyze your machine, looking at a number of various items, such as keypad operation, chip functions, color display (it puts up a beautiful rainbow effect with every one of the 256 colors on the screen), and has the capability to accept machine code. If it finds some discrepancy, it uses a pair of alpha-numeric LEDs to indicate, through a code, where the problem is located. I expect Dick Belton, 4906 Willshire Ave., Baltimore, MD 21206 to have an ad in the next issue.

UV-1 COMPUTER is the commercial version of what we will essentially have when the Add-Under is coupled to the Arcade, except for high-resolution graphics. These units are in use by professionals in the graphic arts, and the current issue of Creative Computing (June) contains an article written by Frank Dietrich and Zsuzsa Molnar describing and illustrating in color some of the graphics that they have obtained with ZGrass using the UV-1 computer. In scanning the article, I believe that those effects could be generated with the Extended Basic without too much difficulty.

A grubby piece of paper was shoved under the door, and on it was a note and a sketch - the note said, in greatest Through the Looking Glass tradition, "TRY ME". The sketch is reproduced here, showing the five resistors that exist just underneath the TV output connector inside the RF modulator. The two outer resistors and the center one are suggested to be cut out entirely, while the remaining two are suggested to be shorted with bits of wire to take them out of the circuit. Supposedly, the output is enhanced. It has helped my set.

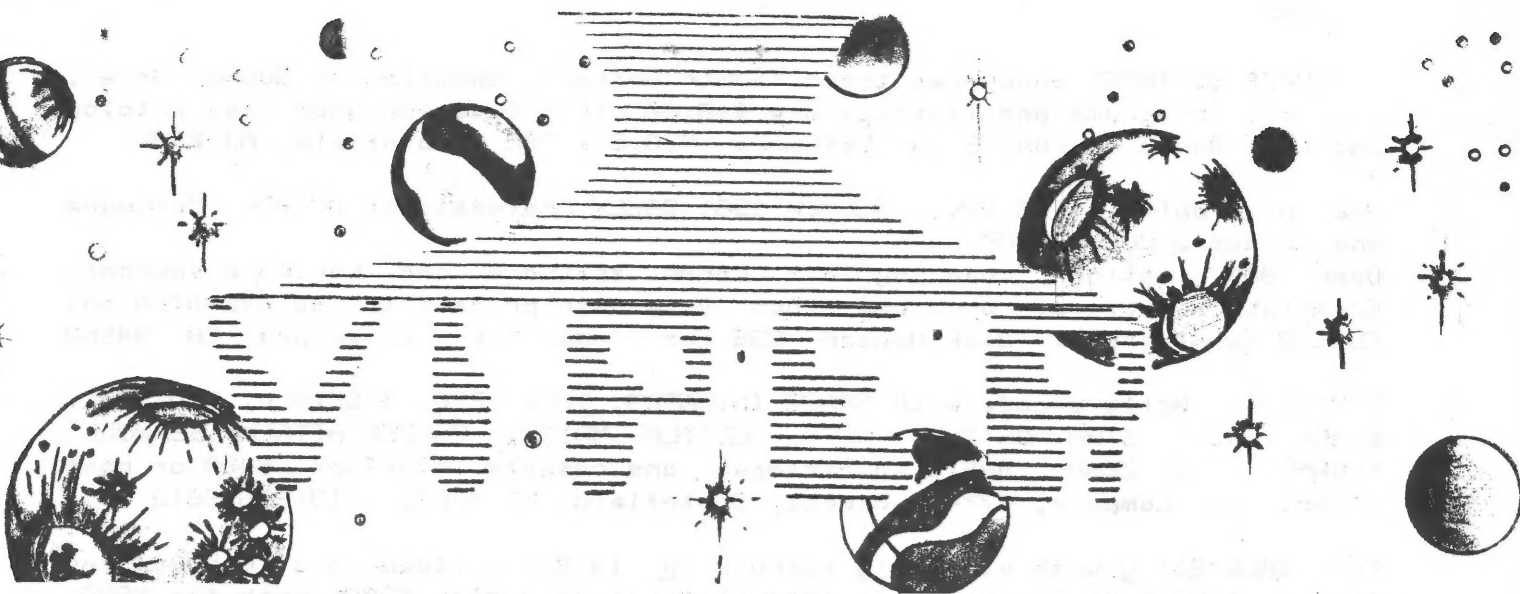


DEFUSE Modifications. Dieter Heinerma sent in the following as a result of some comments and suggestions from Bob Dahl:

```

270 A=RND(100)-1; B=RND(100)-1
280 C=RND(100)-1; IF A>0 GOTO 310
320 PRINT " SIGNAL ",10000-ABS((A+B+C*100)-(D+E+F*100))

```

The VIPER SYSTEM 1 is the first of a series of custom manufactured, quality products made for the Bally Home Computer/Professional Arcade. With this system, you can begin the evolution of your Bally from the Professional Arcade to a powerful graphic computer. The VIPER SYSTEM 1 is a lot more than just a 16K memory expansion. Features and capabilities are listed as follows: one dual position front panel select switch for starting the memory at either 8 or 24K. This will make it convenient for the user to copy any game cartridges and run them in extended RAM where they can be modified or copied to tape. Next, one dual position front panel switch for Auto-Write-Protect or Programmable Write-Protect. The Automatic Write-Protect mode allows the user to load Jay Fenton's excellent new 8K Extended Basic from tape, and then use the Basic to write programs in the remaining 8K. The Programmable-Write-Protect allows you to Write-Protect or Write-Enable the entire 16K RAM board with simple Basic statements. Next, the RAM board located inside the cabinet has two eight position DIP switch packs. Switch pack one enables 4, 8, or 16K bank selectable addressing, and switch pack two controls special bus functions to the Bally and selects either external or internal clocking. SYSTEM 1 also includes a fuse-protected +/- 5 volt and +/- 12 volt power supply. These voltages provide power to the RAM and keyboard interface circuitry. Also included is a heavy duty grounded AC line cord, and filtered AC outlet on the back that is controlled by the front panel switch. The SYSTEM 1 Interface Board provides the bus conversion from the Bally to the VIPER bus, plus a serial keyboard interface which will allow the use of a VIPER or other serialized ASCII keyboard with the system. It also includes one bus cable connecting the VIPER to the Bally. The RAM board and keyboard can be unplugged from the SYSTEM 1 and later plugged directly into the SYSTEM 5 without any changes because the equipment is completely (software and hardware) compatible. For those of you who are home-brewers, when the SYSTEM 1 is upgraded to a SYSTEM 5, the SYSTEM 1 cabinet, power supply, and bus interface card can be used to help prototype your own computer circuits. The entire system is housed in an attractive heavy duty black aluminum cabinet with simulated wood grain side panels and custom silk-screening. The dimensions are 10"x10"x4-1/4".

Due to the response received when originally advertised in past issues of the ARCADIAN, there have been several changes and improvements made to the System RAM card and Interface Card. Therefore, please refer to this advertisement and following advertisements for accurate product information, pricing, and availability. SYSTEM 1 is available now for \$225. A free Extended Basic, on tape and with documentation comes with each SYSTEM 1.

SUPER SOFTWARE announces their "Super Software Spectacular Summer Sale". Now all programs and listings are \$10.00 off!! Send for your free catalog and free Bonus Coupon. Super Software P.O.Box 702 Plainfield, NJ 07061

Now Available... \$5.00... SUMMER 1981 BALLY Professional ARCADE Software and Hardware SOURCEBOOK

Over 300 listings covering all known software and hardware sources. Ccomplete index of programs that have been printed in the ARCADIAN and CURSOR Newsletters. Dick Houser 635 Los Alamos Ave Livermore CA 94550

FOR SALE: Bally arcade with SPACE INVADERS, BALLY PIN, BASEBALL, FOOTBALL, BLACK JACK, STAR BATTLE, MATH, LETTER MATCH, PANZER ATTACK, SEAWOLF, CLOWNS, 280 ZZAP, BASIC cartridges and cassette interface. \$400 or best offer. Jack Lampcov, 27746 Pierce, Southfield, MI 48076 313-599-6818

FOR SALE Bally with virtually everything. 14 Bally Videocades, 6 Wavemaker tapes, 1 year of Cursor tapes, most literature - over \$800 worth for \$500: R.L.Thersen 69 St. Thomas Ct Pleasant Hill CA 94523

COMMUNITY COMPUTERIST's DIRECTORY is similar to a telephone book for locating people or companies having similar interests. The white pages list people with a paragraph of what they are working on, working with, or want to see available. The yellow pages are advertising areas for people, businesses and organizations. July issue will be \$3.50, from P. O. Box 40 Forestville, CA 95436

HAND CONTROLLER REPAIRS With factory prices now at \$45/pair, it pays to have problems fixed - contact Dave Stocker, 333 Coronado Dr., Mt. Vernon, IN 47620

-90-

ARCADIAN

Robert Fabris, pixillated
3626 Morrie Drive
San Jose, CA 95127
The Source, TCD 959

FIRST CLASS
U. S. POSTAGE
PAID

Sunnyvale, CA
Permit No. 931

63301BRG2917H
ARTHUR BERGANTZ
2917 HEADLAND DRIVE
ST CHARLES, MO

63301